

## ASSIGNMENT – 4

### WOKWI AND IBM CLOUD

Assignment Date	17 November 2022
Student Name	Sowmya.P
Student Roll Number	311419106029
Maximum Marks	2 Marks

#### Question-1:

Write code and connections in wokwi for ultrasonic sensor. Whenever the distance is less than 100 cms sent "alert" to IBM cloud and display in device recent events.

#### AIM:

To write code and connections in wokwi for ultrasonic sensor. Whenever the distance is less than 100 cms sent "alert" to IBM cloud and display in device recent events.

#### CODE:

```
#include <WiFi.h> //library for wifi
#include <PubSubClient.h> //library for MQTT

void callback (char* subscribtopic, byte* payload, unsigned int
payloadLength);

//-----credentials of IBM Accounts-----/

//-----
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the
predefined client id by passing parame#define ORG "5xkpdY"
#define DEVICE_TYPE "115devicetype"
#define DEVICE_ID "115deviceid"//Device ID mentioned in ibm watson IOT
Platform
#define TOKEN "authentication" //Token
String data3;
float dist;
//----- Customise the above values -----
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of
event perform and format in which data to be send
```

```

char subscribetopic[] = "iot-2/cmd/test/fmt/String";// cmd REPRESENT command
type AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id by
passing parameter like server id,portand wificredential

int LED = 15;
int trig = 13;
int echo = 12;
void setup()
{
  Serial.begin(115200);
  pinMode(trig,OUTPUT);
  pinMode(echo,INPUT);
  pinMode(LED, OUTPUT);
  delay(10);
  wificonnect();
  mqttconnect();
}
void loop()
{

  digitalWrite(trig,LOW);
  digitalWrite(trig,HIGH);
  delayMicroseconds(10);
  digitalWrite(trig,LOW);
  float dur = pulseIn(echo,HIGH);
  float dist = (dur * 0.0343)/2;
  Serial.print ("Distance in cm");
  Serial.println(dist);

  PublishData(dist);
  delay(1000);
  if (!client.loop()) {
    mqttconnect();
  }
}

/*.....retrieving to
Cloud.....*/

void PublishData(float dist) {
  mqttconnect();//function call for connecting to ibm
  /*
    creating the String in in form JSon to update the data to ibm cloud
  */
  String object;

```

```

if (dist <100)
{
    digitalWrite(LED,HIGH);
    Serial.println("object is near");
    object = "Near";
}
else
{
    digitalWrite(LED,LOW);
    Serial.println("no object found");
    object = "No";
}

String payload = "{\"distance\": ";
payload += dist;
payload += ", \"object\": \"";
payload += object;
payload += "\"}";

Serial.print("Sending payload: ");
Serial.println(payload);

if (client.publish(publishTopic,(char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud
then it will print publish ok in Serial monitor or else it will print publish
failed
} else {
    Serial.println("Publish failed");
}
}

void mqttconnect() {
    if (!client.connected()) {
        Serial.print("Reconnecting client to ");
        Serial.println(server);
        while (!!!client.connect(clientId, authMethod, token)) {
            Serial.print(".");
            delay(500);
        }

        initManagedDevice();
        Serial.println();
    }
}

void wificonnect() //function defination for wificonnect
{
    Serial.println();
}

```

```

    Serial.print("Connecting to ");

    WiFi.begin("Wokwi-GUEST", "", 6); //passing the wifi credentials to establish
the connection
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    Serial.println(WiFi.localIP());
}

void initManagedDevice() {
    if (client.subscribe(subscribetopic)) {
        Serial.println((subscribetopic));
        Serial.println("subscribe to cmd OK");
    } else {
        Serial.println("subscribe to cmd FAILED");
    }
}

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{
    Serial.print("callback invoked for topic: ");
    Serial.println(subscribetopic);
    for (int i = 0; i < payloadLength; i++) {
        //Serial.print((char)payload[i]);
        data3 += (char)payload[i];
    }
    // Serial.println("data: "+ data3);
    // if(data3=="Near")
    // {
    // Serial.println(data3);
    // digitalWrite(LED,HIGH);

    // }

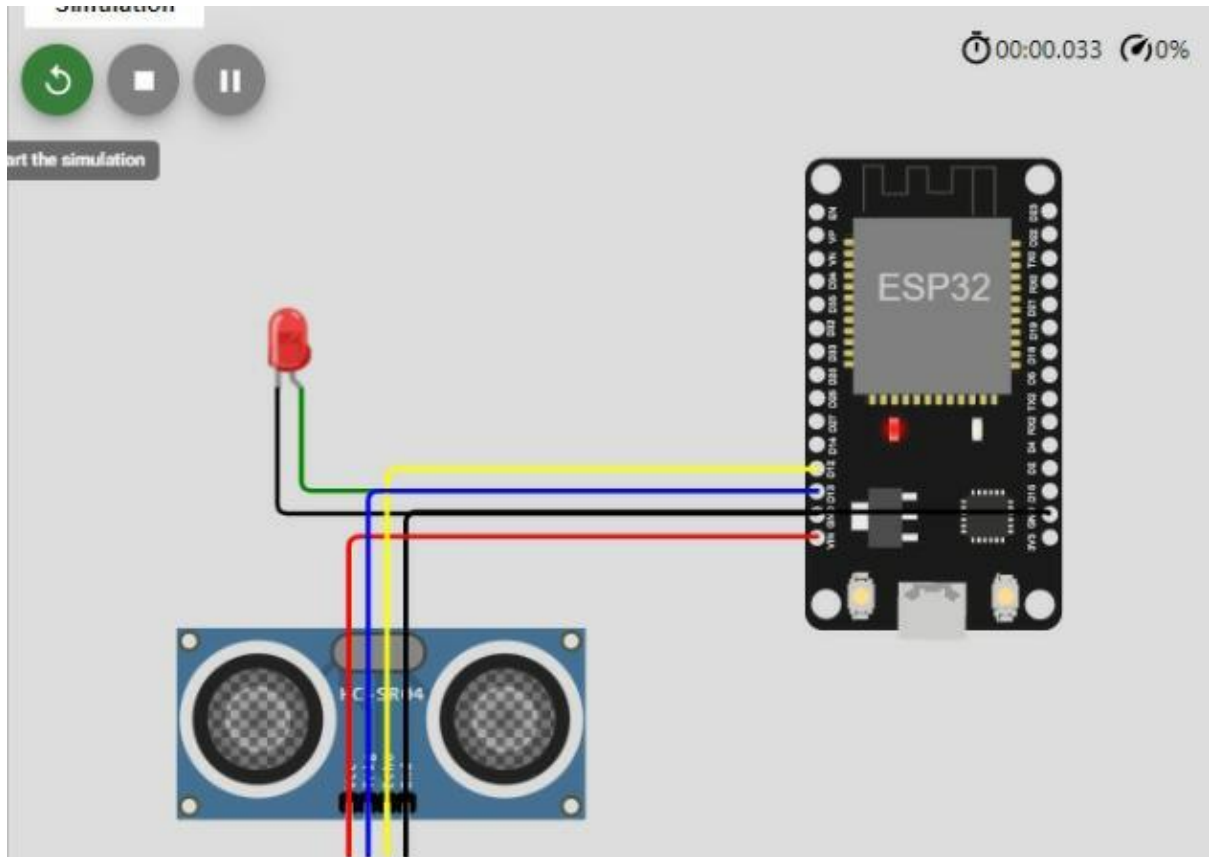
    // else
    // {
    // Serial.println(data3);
    // digitalWrite(LED,LOW);

    // }
    data3="";
}

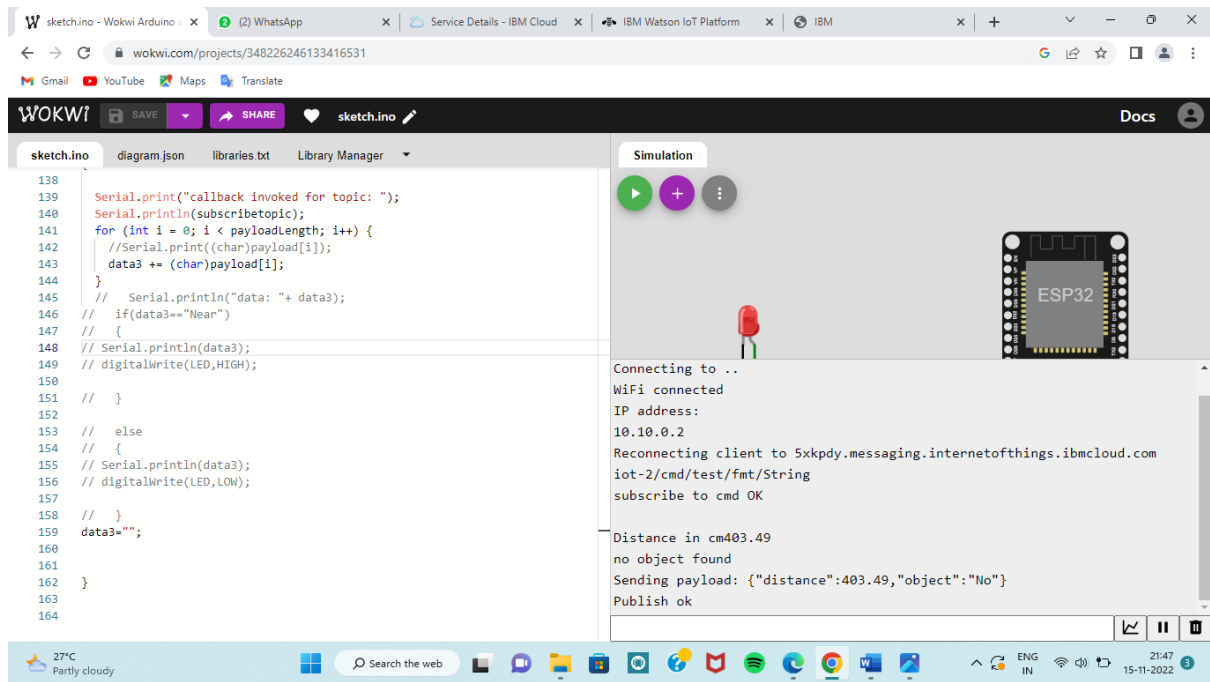
```

}

## CONNECTIONS:



## OUTPUT(WOKWI):



LINK: <https://wokwi.com/projects/348226246133416531>

## OUTPUT (IBM CLOUD):

The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes tabs for 'Browse', 'Action', 'Device Types', and 'Interfaces'. The 'Recent Events' tab is selected, showing a table of live data streams. The table has four columns: 'Event', 'Value', 'Format', and 'Last Received'. The data shows a series of 'Data' events with JSON payloads containing distance and object information. A status box at the bottom right indicates '1 Simulation running'. The bottom of the screen shows a Windows taskbar with various application icons and system information.

IBM Watson IoT Platform

311419106029@smartinternz.com  
ID: 5xkpdv

Browse Action Device Types Interfaces Add Device +

Identity Device Information Recent Events State Logs

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
Data	{"distance":403.47,"object":"","No"}	json	a few seconds ago
Data	{"distance":403.49,"object":"","No"}	json	a few seconds ago
Data	{"distance":403.49,"object":"","No"}	json	a few seconds ago
Data	{"distance":403.51,"object":"","No"}	json	a few seconds ago
Data	{"distance":403.49,"object":"","No"}	json	a few seconds ago

1 Simulation running

27°C Partly cloudy Search the web ENG IN 2151 15-11-2022